

Initial z-vertexing for MAPS+TPC configuration

sPHENIX tracking meeting

Oct 21, 2016

Mike McCumber & Sookhyun Lee

Pull request #190

- Clean up of deprecated functions & variables
- Modularization of PHG4HoughTransformTPC.*
- Initial z-vertexing for MAPS+TPC

Modifications made in

Showing 7 changed files with 2,073 additions and 940 deletions.

Unified Split

 offline/packages/HelixHough/helix_hough/HelixHough.h	+2 -0	
 offline/packages/HelixHough/helix_hough/HelixHough_findHelices.cpp	+25 -1	
 offline/packages/HelixHough/helix_hough/HelixHough_init.cpp	+1 -1	
 offline/packages/HelixHough/helix_hough/sPHENIX/sPHENIXTrackerTPC.cpp	+818 -5	
 offline/packages/HelixHough/helix_hough/sPHENIX/sPHENIXTrackerTPC.h	+33 -0	
 simulation/g4simulation/g4hough/PHG4HoughTransformTPC.C	+1,036 -845	
 simulation/g4simulation/g4hough/PHG4HoughTransformTPC.h	+158 -88	

Initial z-vertexing for MAPS+TPC configuration

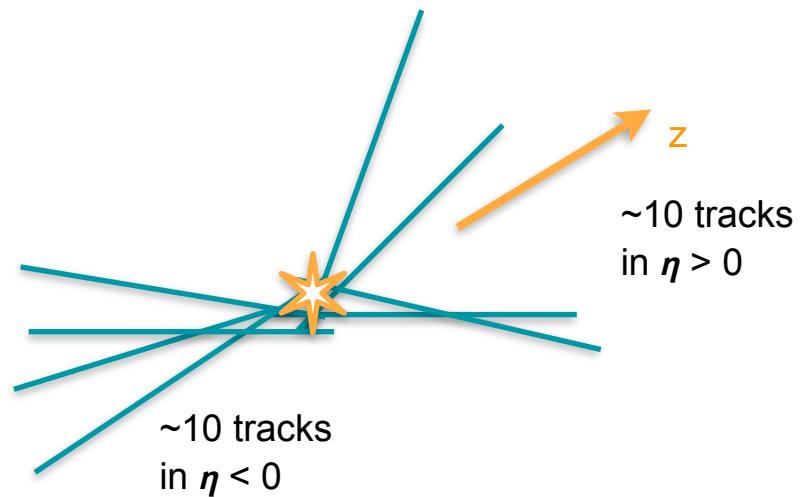
- Currently, the BBC is used to determine initial z-vertex. The BBC offers true z-vertex position info, and does not take into account the uncertainty in reconstructing z-vertex.
- More realistic method would be to use the MAPS+TPC trackers themselves.
- For 7 layer MAPS only configuration, Mike implemented a 3-step z-vertexing method with 7 MAPS layers.

Method for z-vertexing with 7 MAPS layers (by M. McCumber)

Initial step : Get an average over the two z-vertex values obtained from a handful of (~ 10) tracks with positive and negative eta. Target z-vertex resolution = 20 um. (vertex_z : 0.0 -> v_init)

Intermediate step : Use more (~ 40) tracks to get DCA_z resolution better than 2 cm. (vertex_z : v_init -> v_new)

Final step : Full tracking with all reconstructed tracks. Target z-vertex resolution = 5 um. (vertex_z : v_new -> v_final)



Initial z-vertexing with MAPS + TPC trackers

For MAPS+TPC configuration,

- There are much *larger number* of layers (63 instead of 7) in trackers extending to larger radius (80 cm instead of 64 cm).
 - A majority of tracking layers with larger radius in particular is dedicated to '*momentum reconstruction*' rather than '*vertexing*'.
- ➡ A different method than the one used for 7 MAPS layer configuration is needed to get a reasonable z-vertex resolution.

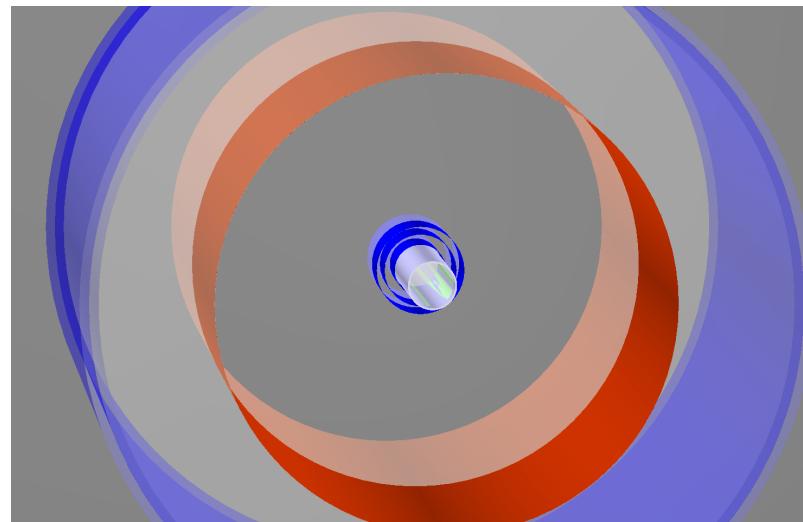
A new method for z-vertexing with MAPS+TPC

- Two-step method

Initial step : Use a few innermost tracking layers, 3 MAPS layers + first 1~2 TPC layers to determine initial z-vertex. Target z-vertex resolution = 20 um. (`vertex_z : 0.0 -> v_init`)

Final step :

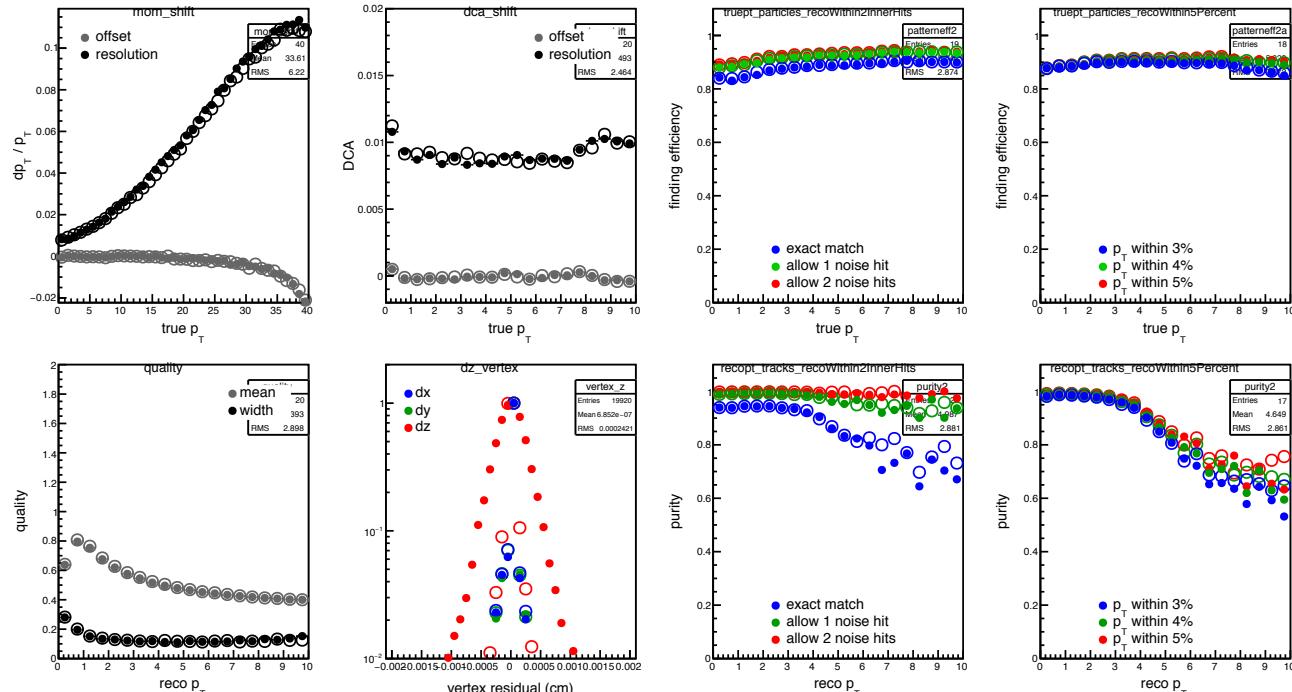
Full tracking with all reconstructed tracks.
Target z-vertex resolution = 5 um.
(`vertex_z : v_init -> v_final`)



initial z-vertexing
using 3 MAPS + 1~2 TPC layers
with `findTracksBySegment` algorithm

Performance tests

- New method for initial z-vertexing with MAPS + TPC (Reference : BBC input (**default**))



100 pions per event,
embedded in
central-Hijing events.

of used layers : 4
initial chisq cut : 3
full chisq cut : 2

- A few hundreds of tracks are needed to get a reasonable z-vertex resolution.

Performance tests

- ➊ Cross-check done for MAPS + IT + TPC configuration with BBC input.
 - ➌ For pure pions as well as embedded pions.
 - : Comparison plots posted on GitHub by Sourav
<https://github.com/sPHENIX-Collaboration/coresoftware/pull/190>
- ➋ Initial z-vertexing with MAPS + IT + TPC with new method
 - ➌ Jobs are running, will update when they are ready

Usage

```
PHG4HoughTransform hough* = new PHG4HoughTransform();  
  
hough->set_use_vertex(true); // use BBC information for initial z-vertexing  
  
hough->set_use_vertex(false); // use tracking detectors to reconstruct z-vertex.
```